

CLAIMS

What is claimed is:

1. A system that facilitates management of a file system filter, comprising:
at least one minifilter that has an altitude associated therewith; and
a filter manager that maps altitudes of the at least one minifilter to legacy filter order groups.
2. The system of claim 1, the altitudes are numeric values.
3. The system of claim 2, the altitudes are unique values.
4. The system of claim 3, the altitudes define the full order of the minifilters with respect to each other.
5. The system of claim 1, wherein multiple instances of the filter manager attach to a file system stack.
6. The system of claim 5, wherein each instance of the filter manager is associated with a unique interval of altitude values.
7. The system of claim 1, the at least one minifilter is coded to permit dynamic loading and/or unloading to a filter stack.
8. The system of claim 7, the altitude of the at least one minifilter ensures that the at least one minifilter, if unloaded, will reload to its previous position in the filter stack.
9. The system of claim 1, further comprising at least one frame dynamically associated with a single minifilter.

10. The system of claim 1, further comprising at least one frame dynamically associated with at least one minifilter.
11. The system of claim 10, further comprising a numerical interval associated with each frame.
12. The system of claim 11, the altitude of at least one minifilter has a value within the numerical interval associated with each frame.
13. A method for managing a file system filter, comprising:
 - loading at least one minifilter to a file system; and
 - determining an altitude associated with the at least one minifilter.
14. The method of claim 13, further comprising scanning at least one filter manager frame in the file system to find an altitude interval [L, H] associated with the at least one filter manager frame, wherein L is the lower boundary value of the interval and H is the upper boundary value of the interval.
15. The method of claim 14, further comprising scanning filter manager frames to determine a frame altitude interval that encompasses the altitude value of the at least one minifilter, such that $L < X < H$, wherein X is the altitude of the at least one minifilter.
16. The method of claim 15, further comprising inserting the at least one minifilter into the filter manager frame with a corresponding altitude interval upon discovery thereof.
17. The method of claim 16, further comprising updating a filter object associated with the at least one minifilter to point to the frame into which the minifilter has been inserted.

18. The method of claim 15, further comprising scanning filter manager frames for altitude intervals, $[L_1, H_1]$ and $[L_2, H_2]$, adjacent to the altitude value X of the at least one minifilter if no single interval $[L, H]$ encompassing the altitude value X of the at least one minifilter is found, such that the value of the altitude, X , of the at least one minifilter is greater than the upper boundary value of the lower interval H_1 and less than the lower boundary value of the higher interval L_2 .
19. The method of claim 18, further comprising:
 - inserting the at least one minifilter into the frame having the higher interval;
 - adjusting the interval of the frame to $[X, H_2]$; and
 - initializing the filter object associated with the at least one minifilter to point to the frame into which the at least one minifilter has been inserted.
20. The method of claim 18, further comprising creating a new frame and stacking the new frame at the top of the file system stack, if no intervals adjacent to the altitude value of the at least one minifilter are found.
21. The method of claim 20 further comprising pre-allocating the new frame for management of the at least one minifilter.
22. The method of claim 21, further comprising calling the filter manager's file system notification routine to submit a request to register for file system notifications.
23. The method of claim 22, further comprising:
 - inserting the minifilter into the new frame;
 - initializing the frame interval upper and lower boundary values to the altitude value of the at least one minifilter such that the interval is $[H, X]$; and
 - updating a filter object associated with the at least one minifilter to point to the new frame;
 - wherein the request to register was successful.

24. The method of claim 22, further comprising:
 - removing the new frame from the filter stack;
 - extracting the altitude interval from the next lower, now top-most, frame in the stack;
 - collapsing the at least one minifilter into the top-most frame; and
 - adjusting the frame interval so that the upper boundary value is set equal to the value of the altitude of the at least one minifilter, such that the adjusted interval is [L, X];

wherein the request for registration failed.
25. The method of claim 22, further comprising determining the identity of a frame calling in to the file system notification routine.
26. The method of claim 25, wherein the identity of the frame is determined by counting the number of all filter manager device objects, N, already in the stack, from top to bottom, using existing application programming interfaces, and wherein each device object represents a frame.
27. The method of claim 26, further comprising initializing a counter to N and decrementing the counter for every node encountered from the bottom to the top of the stack.
28. The method of claim 27, wherein a zero value in the counter represents the position of the frame that corresponds to the attachment of the filter manager.
29. A system that facilitates management of a file system filter, comprising:
 - means for mapping altitudes of minifilters to legacy filter order groups; and
 - means for determining an altitude interval associated with at least one frame.
30. The system of claim 29, further comprising means for inserting at least one minifilter into a frame.

31. The system of claim 30, further comprising means for altering a frame interval to embrace a given minifilter altitude.
32. The system of claim 29, further comprising means for creating a frame for management of at least one minifilter.
34. A computer readable medium having stored thereon the computer executable minifilters and filter manager of claim 1.
35. A computer readable medium having stored thereon computer executable instructions for performing the method of claim 13.